

Amendments to the Claims

Please amend the claims as follows:

1. (currently amended) A method of performing server-side processing of postback input received from a client and associated with a client-side user interface element, the method comprising:

examining the postback input received from the client to determine [[an]] a hierarchical identifier of a target server-side control object in a server-side control hierarchy;

identifying the target server-side control object in the server-side control hierarchy based on the hierarchical identifier of the target server-side control object;

passing the postback input received from the client to the target server-side control object; and

processing the postback input received from the client and passed to the target server-side control object.

2. (original) The method of claim 1 wherein the processing operation comprises changing a property of the target server-side control object; and further comprising:

generating authoring language data from the target server-side control object based on the property to define the client-side user interface element for transmission to the client.

1 3. (currently amended) ~~The method of claim 1~~ A method of performing
2 server-side processing of postback input received from a client and associated with
3 a client-side user interface element, the method comprising:

4 examining the postback input received from the client to determine an
5 identifier of a target server-side control object;

6 identifying the target server-side control object based on the identifier of
7 the target server-side control object;

8 passing the postback input received from the client to the target server-side
9 control object;

10 processing the postback input received from the client and passed to the
11 target server-side control object, wherein the processing operation comprises
12 raising a server-side event from the target server-side control object; and further
13 comprising:

14 generating authoring language data from the target server-side control
15 object based on the server-side event to define the client-side user interface
16 element for transmission to the client.

17 4. (currently amended) The method of claim 2 wherein the postback
18 input is received in a current round trip session, and further comprising:

19 creating a plurality of server-side control objects in a server-side control
20 object hierarchy, responsive to receipt of the postback input for the current round
21 trip session prior to the operation of processing the postback input; and

22 terminating the plurality of server side control objects, prior to receipt of
23 other postback input in a subsequent round trip session after the operation of
24 generating authoring language data.
25

1 5. (original) The method of claim 2 further comprising:
2 searching for the target server-side control object in a server-side control
3 hierarchy based on the identifier;
4 creating the target server-side control object in the server-side control
5 hierarchy, if the target server-side control object is not found by the searching
6 operation; and
7 terminating the server-side control hierarchy, after the operation of
8 generating authoring language data.

9 6. (currently amended) The method of claim 1 wherein the hierarchical
10 identifier has a hierarchical identifier structure indicating a plurality of levels in a
11 server-side control ~~object~~ hierarchy including a plurality of member server-side
12 control objects, and the operation of identifying the target server-side control
13 object comprises:

14 extracting a node level identifier from the hierarchical identifier;
15 passing the node level identifier to a member server-side control object
16 corresponding to the node level identifier;

17 identifying the member server-side control object as the target server-side
18 control object, if the node level identifier identifies a leaf node of the hierarchical
19 identifier;

20 extracting a next node level identifier from the hierarchical identifier of the
21 target server-side control object, if the node level identifier does not identify a leaf
22 node of the hierarchical identifier, wherein the next node level identifier identifies
23 a child server-side control object of the member server-side control object; and
24

1 performing recursively the passing and identifying operations and the
2 operation of extracting a next node level identifier using the next node level
3 identifier as the node level identifier and the child server-side control object as the
4 member server-side control object, if the node level identifier does not identify a
5 leaf node of the hierarchical identifier.

6 7. (original) The method of claim 1 wherein the operation of
7 processing the postback input comprises:

8 storing a postback data value as a property the target server-side control
9 object.

10 8. (original) The method of claim 7 wherein the target server-side
11 control object initially stores an old data value as a property, and the operation of
12 storing a postback data value comprises:

13 associating the postback data value with the property;

14 indicating a data change associated with the target server-side control
15 object, if the postback data value passed to the target server-side control object is
16 different than the old data value of target server-side control object; and
17

18 replacing the old data value with the postback data value in the target
19 server-side control object.

20 9. (original) The method of claim 8 further comprising:

21 raising a server-side data change event after the operation of replacing of
22 the old data value, if a data change is indicated.

23 10. (original) The method of claim 7 wherein the target server-side
24 control object is one of a plurality of member server-side control objects in a
25

1 server-side control object hierarchy, and the operation of storing a postback data
2 value comprises:

3 storing postback data values for all of the member server-side control
4 objects in the server side control object hierarchy; and

5 raising at least one server-side data change event after the operation of
6 storing a postback data value for all member server-side control objects, if at least
7 one data change is indicated.

8 11. (original) The method of claim 10 further comprising:

9 receiving the server-side data change event from the target server-side
10 control object; and

11 invoking a function of a non-user-interface server component, based on the
12 server-side data change event.

13
14 12. (original) The method of claim 1 wherein the operation of
15 processing the postback input comprises:

16 processing a postback event using the target server-side control object.

17 13. (original) The method of claim 12 wherein the operation of
18 processing a postback event comprises:

19 extracting from the postback input a postback event argument associated
20 with the identifier;

21 passing the postback event argument associated with the identifier to the
22 target server-side control object; and

23 processing the postback event argument using the target server-side control
24 object.
25

1 14. (currently amended) ~~The method of claim 13~~ A method of
2 performing server-side processing of postback input received from a client and
3 associated with a client-side user interface element, the method comprising:

4 examining the postback input received from the client to determine an
5 identifier of a target server-side control object;

6 identifying the target server-side control object based on the identifier of
7 the target server-side control object;

8 passing the postback input received from the client to the target server-side
9 control object; and

10 raising a server-side event from the target server-side control object,
11 responsive to the operation of passing the postback input. ~~processing the postback~~
12 ~~event argument.~~

13 15. (original) The method of claim 14 further comprising:

14 receiving the server-side event from the target server-side control object;

15 and

16 invoking a function of a non-user-interface server component, based on the
17 server-side event.

18
19 16. (currently amended) A computer data signal embodied in a carrier
20 wave by a computing system and encoding a computer program for executing a
21 computer process performing server-side processing of postback input received
22 from a client and associated with a client-side user interface element, the computer
23 process comprising:

1 examining the postback input received from the client to determine [[an]] a
2 hierarchical identifier of a target server-side control object in a server-side control
3 hierarchy;

4 identifying the target server-side control object in the server-side control
5 hierarchy based on the hierarchical identifier of the target server-side control
6 object;

7 passing the postback input received from the client to the target server-side
8 control object; and

9 processing the postback input received from the client and passed to the
10 target server-side control object.

11 17. (currently amended) A computer program storage medium readable
12 by a computer system and encoding a computer program for executing a computer
13 process performing server-side processing of postback input received from a client
14 and associated with a client-side user interface element, the computer process
15 comprising:

16 examining the postback input received from the client to determine [[an]] a
17 hierarchical identifier of a target server-side control object in a server-side control
18 hierarchy;

19 identifying the target server-side control object in the server-side control
20 hierarchy based on the hierarchical identifier of the target server-side control
21 object;

22 passing the postback input received from the client to the target server-side
23 control object; and
24
25

1 processing the postback input received from the client and passed to the
2 target server-side control object.

3 18. (canceled)

4
5 19. (currently amended) ~~The computer program product of claim 18~~
6 ~~wherein the computer process further comprises:~~ A computer program product for
7 processing postback input received from a client and associated with a client-side
8 user interface element, the computer process comprising:

9 receiving ~~[[an]]~~ a hierarchical identifier associated with the target server-
10 side control object; ~~[[and]]~~

11 creating a plurality of server-side control objects in a server-side control
12 hierarchy, the server-side control hierarchy including a target server-side control
13 object associated with the client-side user interface element;

14 identifying the target server-side control object within the server-side
15 control ~~object~~ hierarchy, based on the hierarchical identifier~~[[.]]~~;

16 passing the postback input to the target server-side control object;

17 processing the postback input received by target server-side control object;

18 and

19 generating authoring language data from the plurality of server-side control
20 objects to define a web page for display on a client.

21 20. (currently amended) The computer program product of claim ~~[[18]]~~
22 19 wherein the computer process further comprises:

23 terminating the plurality of server side control objects, after the operation of
24 generating authoring language data.
25

1 21. (canceled)

2 22. (canceled)

3 23. (canceled)

4 24. (previously presented) A computer program product embodied in a
5 computer readable medium for executing a computer process, the computer
6 process comprising:
7

8 generating authoring language data from a plurality of server-side control
9 objects at a server to define a page for display on a client, the authoring language
10 data including a script that is tagged to be executed by the server to process input
11 data received from the client.

12 25. (previously presented) The computer program product of claim 24,
13 wherein each server-side control object corresponds to a client-side user interface
14 element.
15

16 26. (previously presented) A method comprising:
17 generating authoring language data from a plurality of server-side control
18 objects at a server to define a web page for display on a client, the authoring
19 language data including a script that is tagged to be executed by the server to
20 process input data received from the client.

21 27. (currently amended) A method comprising:
22 receiving at a server input data from [[the]] a client ~~at the server~~, the input
23 data being associated with an individual client-side user interface element on the
24 client;
25

1 generating a hierarchy of server-side control objects on [[a]] the server,
2 each server-side control object corresponding to a client-side user interface
3 element on [[a]] the client;

4 identifying, using a hierarchical identifier received from the client, a server-
5 side control object in the hierarchy to which the individual client-side user
6 interface element corresponds; and

7 processing the input data using the identified server-side control object.

8 28. (previously presented) The method of claim 27 wherein the
9 processing operation comprises:

10 setting a property value of the identified server-side control object based on
11 the input data.

12 29. (previously presented) The method of claim 27 wherein the
13 processing operation comprises:

14 raising an event in the identified server-side control object based on the
15 input data.

16 30. (previously presented) The method of claim 27 further comprising:
17 generating authoring language data from a plurality of server-side control
18 objects at a server to define a page for display on the client, the page including the
19 individual client-side user interface element.

20 31. (previously presented) The method of claim 27 further comprising:
21 generating authoring language data from a plurality of server-side control
22 objects at a server to define a page for display on the client, the authoring language
23
24
25

1 data including a script that is tagged to be executed by the server to process input
2 data received from the client.

3 32. (currently amended) A computer program product embodied in a
4 computer readable medium for executing a computer process, the computer
5 process comprising:

6 receiving at a server input data from [[the]] a client ~~at the server~~, the input
7 data being associated with an individual client-side user interface element on the
8 client;

9 generating a hierarchy of server-side control objects on [[a]] the server,
10 each server-side control object corresponding to a client-side user interface
11 element on [[a]] the client;

12 identifying, using a hierarchical identifier received from the client, a server-
13 side control object in the hierarchy to which the individual client-side user
14 interface element corresponds; and

15 processing the input data using the identified server-side control object.

16
17 33. (previously presented) The computer program product of claim 32
18 wherein the processing operation comprises:

19 setting a property value of the identified server-side control object based on
20 the input data.

21 34. (previously presented) The computer program product of claim 32
22 wherein the processing operation comprises:

23 raising an event in the identified server-side control object based on the
24 input data.
25

1 35. (previously presented) The computer program product of claim 32
2 wherein the computer process further comprises:

3 generating authoring language data from a plurality of server-side control
4 objects at a server to define a page for display on the client, the page including the
5 individual client-side user interface element.

6 36. (previously presented) The computer program product of claim 32
7 wherein the computer process further comprises:

8 generating authoring language data from a plurality of server-side control
9 objects at a server to define a page for display on the client, the authoring language
10 data including a script that is tagged to be executed by the server to process input
11 data received from the client.

12
13 37. (canceled)
14
15
16
17
18
19
20
21
22
23
24
25